(The following statement was delivered to the Joint Committee on Election Laws of the Massachusetts General Court on 19 July 2005.)

I hold a Master's degree in Computer Science from MIT and have over 30 years experience in programming and computer systems consulting, most recently in wireless and network security. I am retired from Hewlett Packard and am now a principal in a software startup.

The word "machine" is used to refer to any computer-based election systems, including Direct Recording Electronic ("DRE" -- commonly also called "touch screens"), Optical Scan, and central tabulating systems.

Software IS a Problem

As a computer professional, I find myself in a bit of an awkward position arguing against the use of computers in elections. My position is this: computers are extremely helpful, even necessary, to solving many problems; but computers are not essential to elections, and the risks are just too great.

We would be unable to use our ATM cards to access our bank accounts from around the world without computer-based funds transfers. However, there are risks associated with all those computer systems and transmission links carrying all those funds transfers, and extraordinary measures are taken to avoid the risks, and to detect tampering or other breaches of security. For modern financial transactions, there is no alternative to the use of computer-based systems.

Computer based systems are not essential to the conduct of elections. Many large democracies in the developed and developing world conduct their elections without computer-based systems. There are basically only three real benefits to the use of

computers in elections. One is that results are available a few hours earlier. The second is that certain accomodations can be made for voters with disabilities. The third is that certain errors made by voters in marking their ballots can be detected and the voter informed so that a correction may be made at the polling place. Getting early results is an extremely minor benefit that must be weighed against the dangers, which I will outline below. The other benefits, accessibility and checking for voter errors, can be provided without using computer systems in the *counting* and *tabulating* process.

Software is a powrful medium for solving problems. Software can be duplicated easily and instantly. Software is what makes a computer-based system perform its functions. Anything that a computer-based system can do is performed at the command of the software running invisibly inside, perhaps transmitted the instant before from somewhere else.

As a result, *software is a powerful medium for creating problems.* A software defect can cause any kind of malfunction. Both pranksters and saboteurs love to work with software. Malicious software can take advantage of phone lines and networks and memory cards and discs to transmit itself to other systems. Malicious software can lie in wait -- even for years -- before doing its evil deeds. Malicious software can cover its tracks and even erase itself after the deed is done.

In my work as a computer systems consultant, I must assume that attempts will be made to attack, compromise, and invade any software-based systems I design. I must be humble enough to assume that a clever prankster or saboteur may overcome my best defenses. As a result I design systems to check both for innocent errors that WILL occur and deliberate tampering. I must always check for intrusions and failures, and the system must be designed

so that *reliable independent and original records* are maintained so that a meaningful check can be made.

It Only Takes One Person

One person acting alone can cause many computer-based machines to malfunction.

One person can write a piece of software (a "virus" or a "Trojan horse" -- we'll call them generically an "intrusion"') that can corrupt any number of machines. A machine can be infected at any time before an election. Software can even be infected before it is put on the machine, even at the factory.

Intentional sabotage (by an authorized programmer) is also always a possibility. Consider that the software is held to be a "secret" by the vendors; this possibility cannot be dismissed.

Well-intentioned programmers sometimes make provisions in the software for "maintenance"; while not directly malicious, such provisions can subsequently be exploited to alter the software in malicious ways.

Any connection, permanent or temporary, can be exploited to transmit an intrusion. By "connection" I mean a computer network, a phone line, a memory card, a disk, or wireless communication to an internal device. (Note that the person actually establishing the connection, for example, inserting a card, may not know that a software intrusion is being transmitted -- as far as they know, it is an innocent maintenance or data retrieval operation.)

Given the attraction and high value of election tampering, we must assume that tampering will be attempted, and that it may sometimes succeed in spite of our best efforts. Thus if we were to use computer-based systems we would have to take measures to

detect tampering with election machines, and we would have to implement procedures to *recover* from this tampering. However, as I will show, tampering is surprisingly hard to detect, even harder to prove, and after-the-fact recovery mechanisms may be ignored.

It Only Takes One Vote (per Machine)

In 2004, Ohio didn't appear to be that close, but a shift in just one vote in 87 would have changed the outcome of the presidential race in Ohio and, thus, in the US. This would need only a handful of changes per machine.

A change in many machines is as easy as a change in one machine. Thus the pattern we are more likely to experience -- but less likely to notice -- is one of many small discrepancies on many machines.

Who would do this? This tampering could be accomplished by a single individual or a small band -- a prankster, a disgruntled employee, an unscrupulous campaign worker, a vendor that is over-zealous in its support of a candidate, organized crime, a foreign power, or a terrorist group -- anyone with an interest in or desire to see a particular outcome in any US election, or perhaps just wanting to create chaos.

Tampering is hard to detect

Software in a machine is hard to see, and hard to fully understand, even for experts! Software intrusions can accomplish any effect; in particular they can mimic "glitches" and human error.

Since many machines can be infected, and since only a small change in result is needed per machine, the tampering is easy to miss or overlook. For example, someone can switch whom votes are for, but keep the total number of votes cast the same. This kind

of insidious small change is easy to ignore, or easy to dismiss as "insignificant."

There were tens of thousands of reported small computer problems in 2004. But we don't know how many additional problems were never reported because they were not noticed or they were considered "insignificant."

Some kinds of tampering might look quite harmless -- for example, an occasional "default" vote (which has the side effect of a "higher quality" election -- fewer "undervotes"!)

Another kind of "innocent" tampering is one that doesn't alter votes and thus cannot be detected by any kind of auditing. The election can be biased against certain precincts by software tampering that causes the machines in those areas to slow down or crash. If these precincts are chosen to be precincts that favor one particular candidate or party, such tampering will cause that candidate or party to lose votes. *This is why I discourage any thought of "auditing" and "paper trails" as solutions to the threats against electronic voting.*

We cannot assume that fraud would be "obvious" if it were serious enough to change the outcome of an election. Software intrusions can cover their tracks, even erase themselves when done -- only the altered election result remains!

Can machines be made more secure?

The very nature of computer-based systems makes the above risks possible -- one person making very small changes in many places without leaving "tracks" is just not possible with paper!

Today's computerized voting systems are very poorly designed with regards to security -- passwords are widely known and are rarely changed, breakable forms of encryption are used, and

systems are connected to networks, phone lines, and memory devices without "best practices" in security. Once they are delivered, election systems are rarely under tamper-proof seal from the point at which known certified software is loaded.

Regarding certification and testing: it is a maxim in computer science: *"Testing can only show the presence of errors, never the absence of errors."* Likewise, testing cannot prove the absence of malicious code or the absence of opportunities for intrusion. Testing the software is not a solution.

Some of the problems with computer-based systems have technological fixes, but only at the cost of increased complexity, rendering the systems beyond the knowledge of all but a handful of experts. All of us non-experts would simply have to trust that these systems had not been compromised.

A quote from computer science Professor David L. Dill of Stanford University, is sums up the problem quite well:

> *Why am I always being asked to prove these systems aren't secure? The burden of proof ought to be on the vendor. You ask about the hardware. 'Secret.' The software? 'Secret.' What's the cryptography? 'Can't tell you because that'll compromise the secrecy of the machines.'... Federal testing procedures? 'Secret'! Results of the tests? 'Secret'! Basically we are required to have blind faith.*

I can assure you, even if *nothing* were secret, it would still be a practical impossibility to *prove* the security and reliability of a state-of-the-art electronic voting machine.

On paper trails and auditing

Note that in systems that print a "paper trail", the paper trail itself is created by software that may be altered by tampering or error,

and thus is unreliable as a record. Having the voter review the paper trail is an attempt to fix this additional problem, but it is an attempt that is likely to fail.

A "voter verified paper audit trail" is a problematic attempt to create the equivalent of an original document -- using, in part, the system being audited to create its audit document! That document itself must then be "audited" by the voter. Such an "audit trail" is certain to be an accurate reflection of what the voters selected only if 100% of voters check 100% of the votes 100% correctly -- an impossibility in real situations. Otherwise we start out with an audit record that itself cannot be assumed to be 100% correct, resulting in a less than useless sham of an audit.

With a printed paper trail we also have the problem of "what if we find a discrepancy"? If we only see one or two discrepancies per machine, would we do anything about it? Would it just be treated as a glitch, written down and forgotten? Would that one machine be taken out of service -- but what about the votes it already "counted"? What about the other, presumably similar machines on which no voter reported a discrepancy? Remember, all it takes to steal an election are a few discrepancies per machine.

The "Political Realism" Problem

The first "official" results create a strong presumption of the correct result. How compelling would any evidence of tampering have to be to work against that presumption? Our efforts must be directed towards limiting the opportunities for tampering in the first place. If possible, the counting process itself should produce the first crosscheck or audit of the result. One way to accomplish this is to immediately count the ballots twice by two independent teams. Detection of tampering is always necessary, but experience shows that evidence of tampering won't always change a tampered result if it is delayed.

Simplicity, Transparency, Openness

Our voting systems must be simple enough so that non-technical observers can see what is going on. They must be transparent and open enough so that, once the ballot is cast in secret, the rest of the process is observable by the public, and all intermediate results are open to checking by all. Our election systems must be designed so that the secret actions of a few cannot have an effect without raising suspicion. (It is unfortunate that in some jurisdictions you will be arrested if you try to observe the vote counting process.)

Paper systems can be made to meet these criteria easily; computer-based systems cannot. When this fact is combined with recent results that show a lower error rate for hand-counted paper ballots, is there any reason to consider machines?

Only a hand-marked paper ballot is an original documentation of the voter's intent. We must have that at a minimum, and if we have that, there is no reason against (and many reasons for) counting those ballots in an open, public process that is visible and understandable to all.

Software based systems introduce many opportunities for problems, including tampering. "One person, one vote" must be a principle of democracy, not a description of all it takes to steal an election!

Bob Fleischer
119 Nashua Rd
Groton, MA 01450
(978) 448-6151
AOL IM: rjf7r
www.caef.us